

Jalankan program tersebut dan buka <http://localhost:8080/student/view/12345> dan <http://localhost:8080/student/view/15432> (nomor student disesuaikan dengan database masing-masing).

View Student by NPM	View Student by NPM	View Student by NPM
localhost:8080/student/view/12345	localhost:8080/student/view/15432	localhost:8080/student/view/12344
NPM = 12345	NPM = 15432	NPM = 12344
Name = chaneki	Name = dengklek	Name = ekki
GPA = 3.45	GPA = 3.99	GPA = 2.75
Kuliah yang diambil	Kuliah yang diambil	Kuliah yang diambil
• MPKT-6 sks	• PSP-4 sks	• SDA-3 sks

LATIHAN

- Ubah method selectAllStudents pada kelas **StudentMapper** agar halaman viewall menampilkan semua student beserta daftar kuliah yang diambil.

View All Students
localhost:8080/student/viewall
<div> Delete Data Update Data </div>
<div> <div> <div>All Students</div> <div> <div> <div>No. 1</div> <div>NPM = 12344</div> <div>Name = ekki</div> <div>GPA = 2.75</div> <div>Kuliah yang diambil</div> <div>• SDA-3 sks</div> </div> <div> <div>No. 2</div> <div>NPM = 12345</div> <div>Name = chaneki</div> <div>GPA = 3.45</div> <div>Kuliah yang diambil</div> <div>• MPKT-6 sks</div> </div> <div> <div>No. 3</div> <div>NPM = 15432</div> <div>Name = dengklek</div> <div>GPA = 3.99</div> <div>Kuliah yang diambil</div> <div>• PSP-4 sks</div> </div> </div> </div> </div>

- Buatlah view pada halaman <http://localhost:8080/course/view/{id}> untuk Course sehingga dapat menampilkan data course beserta Student yang mengambil.

View Student by NPM	View Student by NPM	View Student by NPM
localhost:8080/course/view/CSC126	localhost:8080/course/view/CSC123	localhost:8080/course/view/CSC124
ID = CSC126	ID = CSC123	ID = CSC124
Nama = MPKT	Nama = PSP	Nama = SDA
SKS = 6	SKS = 4	SKS = 3
Mahasiswa yang mengambil	Mahasiswa yang mengambil	Mahasiswa yang mengambil
• 12345 - chaneki	• 15432 - dengkleki	• 12344 - ekki

Jelaskan apa saja hal yang Anda pelajari dari tutorial ini.

- Pada tutorial kali ini, saya belajar untuk menampilkan data lebih dari satu tabel dengan melakukan join dengan tabel lain

Cantumkan juga penjelasan Anda terhadap hal-hal berikut:

- Method yang Anda ubah pada Latihan Merubah SelectAllStudents , jelaskan

```
@Select("select npm, name, gpa from student")
@Results(value = { @Result(property = "npm", column = "npm"), @Result(property = "name", column = "name"),
    @Result(property = "gpa", column = "gpa"),
    @Result(property = "courses", column = "npm", javaType = List.class, many = @Many(select = "selectCourses")) })
List<StudentModel> selectAllStudents();
```

- Method yang Anda buat pada Latihan Menambahkan View pada Course, jelaskan
Menambahkan kode tersebut pada kelas StudentMapper, untuk mengambil atribut pada tabel course, kemudian melakukan join untuk mendapatkan atribut pada tabel student

```
@Select("select id_course, name, credits from course where id_course = #{id}")
@Results(value = { @Result(property = "id", column = "id_course"), @Result(property = "name", column = "name"),
    @Result(property = "credits", column = "credits"),
    @Result(property = "students", column = "id_course", javaType = List.class, many = @Many(select = "selectStudents")) })
CourseModel selectCourse(@Param("id") String id);

@Select("select student.npm, name, gpa " + "from studentcourse join student "
    + "on studentcourse.npm = student.npm " + "where studentcourse.id_course = #{id}")
List<StudentModel> selectStudents(@Param("id") String id);
```

Menambahkan kode tersebut pada kelas StudentService

```
CourseModel selectCourse(String id);
```

Menambahkan method tersebut pada kelas StudentServiceDatabase

```
@Override
public CourseModel selectCourse(String id) {
    Log.info("select course with id {}", id);
    return studentMapper.selectCourse(id);
}
```

Menambahkan requestmapping pada kelas StudentController

```
@RequestMapping("/course/view/{id}")
public String cPath(Model model, @PathVariable(value = "id") String id) {
    CourseModel course = studentDAO.selectCourse(id);

    if (course != null) {
        model.addAttribute("course", course);
        return "view2";
    } else {
        model.addAttribute("id", id);
        return "not-found2";
    }
}
```

Membuat file view2.html dan not-found2.html

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
<title>View Student by NPM</title>
</head>
<body>
<h3 th:text="'ID = ' + ${course.id}">Course ID</h3>
<h3 th:text="'Nama = ' + ${course.name}">Course Name</h3>
<h3 th:text="'SKS = ' + ${course.credits}">Course Credits</h3>
<h3>Mahasiswa yang mengambil</h3>
<ul th:each="student, iterationStatus: ${course.students}">
<li th:text="${student.npm} + ' - ' + ${student.name}">
    NPM - Nama</li>
</ul>
</body>
</html>
```

```
<!DOCTYPE html>
<html xmlns:th="http://www.thymeleaf.org">
<head>
<title>Not Found</title>
</head>
<body>
<h1>Course not found</h1>
<h3 th:text="'ID = ' + ${id}">Course ID</h3>
</body>
</html>
```